

Concepto de WEB Stack

De acuerdo con Ionos Digital Guide, un *web stack*, también denominado *web application stack* o conjunto de soluciones (*solution stack*), define a un paquete de software necesario para el desarrollo de páginas y aplicaciones web. Pueden ser openSource (soporte de una comunidad) o propietarias (como Adobe o Microsoft, donde el soporte es ofrecido por las compañías).

El stack tradicional usado como infraestructura para páginas y aplicaciones web se compone, como mínimo, de los cuatro componentes de software:

Un sistema operativo	Es la interfaz central entre el hardware fundamental del servidor y los otros componentes de software.
Un servidor web	<p>El servidor web es un software instalado en el sistema operativo que entrega documentos al cliente que los solicita.</p> <p>La solicitud del cliente es comunicada al servidor web mediante el protocolo de transmisión HTTP.</p> <p>Si estos documentos solicitados son contenidos estáticos como archivos HTML o imágenes, el servidor web procesa esta petición directamente entregándolos al cliente, pero el contenido dinámico, por el contrario, solo puede transmitirlo con ayuda de módulos de programación.</p>
Una base de datos	<p>En los sistemas de bases de datos, se almacenan, de forma eficiente y a largo término, los datos necesarios para que el servidor web, en el momento requerido, pueda recurrir a ellos.</p> <p>Para ello, el servidor emite peticiones con ayuda de las extensiones del servidor, que la base de datos procesa y devuelve a las extensiones del servidor.</p> <p>La elección de la base de datos está estrechamente ligada al software del servidor web y al lenguaje de programación utilizado.</p>
Un intérprete de lenguaje de programación	Los intérpretes de lenguaje actúan, por un lado, en la parte del cliente (en el navegador que realiza la petición) y, por otro, la programación en la parte del servidor hace posible los cálculos necesarios para representar las páginas.

Consideraciones al escoger una stack.

Escalabilidad: tanto horizontal y vertical.

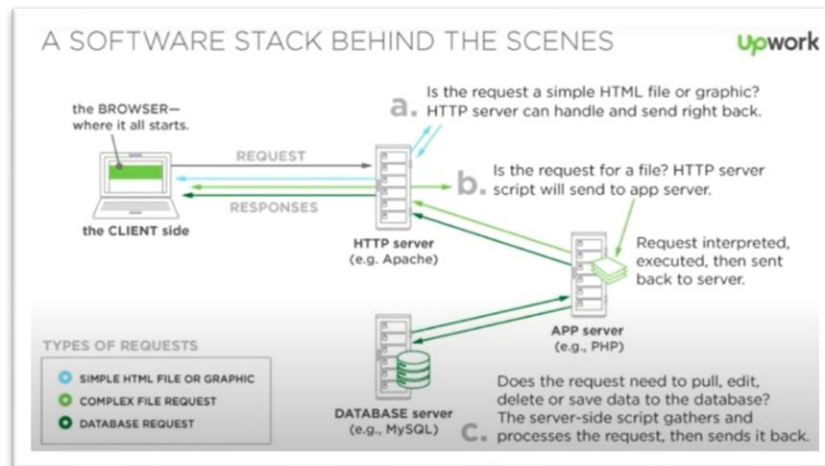
¿Cuáles son las fortalezas de los desarrolladores en la selección de las tecnologías?

¿Qué tan importante es la velocidad?

¿El costo es un factor importante?

¿Cómo funciona un software web internamente?

En la figura 1 se ejemplifica el proceso de atención de una petición generada desde un browser y cómo entran en juego cada una de los componentes del webstack seleccionado.



Algunos ejemplos de WEB Stack

MEAN Stack

- relative modern stack
- Write an entirely application with just Javascript Language
- Technologies
 - M: MongoDB
 - NoSQL vs SQL
 - E: Express
 - A: Angular(js, 2/4/5)
 - N: Nodejs



.NET Stacks

- WISA
 - Windows (OS)
 - IIS (Application Server)
 - SQL Server (database)
 - ASP.NET (Server side Language)



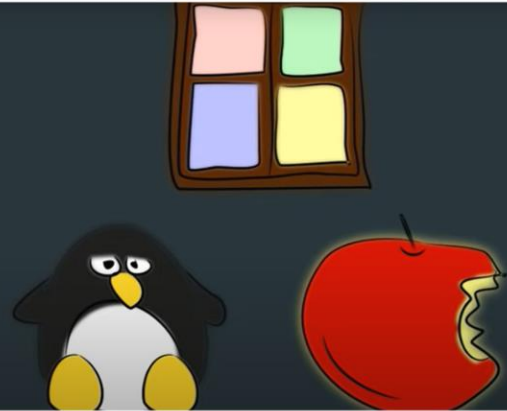
LAMP

- One of the First most popular Stacks
- made all for free, and open source
- Technology:
 - L: Linux Distributions
 - A: Apache Server
 - M: MySQL
 - P: python, perl, php
- Benefits
 - flexible, customizable
 - easy to develop and deploy



LAMP Variations

- WAMP: Windows Server Dist
- WIMP: Windows and IIS server
- LAPP: Postgre instead of Mysql
- MAMP: Mac OSX instead of Linux



Javascript Stacks Variations

- MERN: React instead of Angular
- MEVN: Vuejs instead of Angular
- MEBN: backbone
- MEEN: Ember
- SENA: SQL
 - Mysql
 - MS SQL
 - Oracle
 - Postgre





EQUIPO No. 1	INTEGRANTES: Leonardo Céspedes Tenorio c.2022080602 Kevin Chang Chang c.2022039050 Frankmin Feng Zhong c.2022089248 Erika Cerdas Mejias c.2022138199
Selección del <i>web stack</i> para el proyecto de diseño	<p>WAMP (Windows, Apache, MySQL, PHP)</p> <p>Windows será el sistema operativo host donde se llevarán a cabo las actividades de desarrollo.</p> <p>Apache desempeñará el rol de servidor HTTP de código abierto.</p> <p>MySQL actuará como el sistema de gestión de bases de datos relacionales (RDBMS) encargado de almacenar y administrar de manera eficiente todos los datos requeridos por la aplicación web.</p> <p>PHP será el lenguaje de programación utilizado para el desarrollo del lado del servidor (backend).</p> <p>SASS (Syntactically Awesome Stylesheets) se usará como preprocesador de CSS, el cual brinda numerosas ventajas y funcionalidades adicionales.</p> <p>Gulp será utilizado como sistema de automatización de tareas y compilación en streaming.</p> <p>Composer se empleará como gestor de dependencias (paquetes) para el desarrollo en PHP.</p>
Justificación	En cuanto a la experiencia de los desarrolladores en este web stack, tres integrantes tienen conocimiento básico-regular en este, ya que fue utilizado previamente

	<p>en un único proyecto de universidad. El otro integrante tiene más experiencia ya que ha desarrollado otros proyectos personales utilizándolo. También, se trata de un web stack que permite una instalación y configuración sencilla y rápida de todas las herramientas necesarias para el desarrollo. Se utilizará una arquitectura de tipo MVC, el lenguaje de programación que será utilizado (PHP) permite una eficiente y sencilla comunicación entre backend y frontend. En cuanto a la escalabilidad: WAMP ofrece buena flexibilidad y adaptabilidad ante proyectos de distintos tamaños y complejidades: Apache permite el balanceo de carga entre múltiples servidores web, MySQL admite la replicación de datos y el particionamiento, todas las herramientas se pueden implementar en hardware más potente para mejorar el rendimiento.</p> <p>Respecto a la velocidad, Apache es un servidor web rápido y optimizado, MySQL es un gestor de bases de datos de alto rendimiento, PHP, aunque no es tan rápido como otros lenguajes, tiene un rendimiento aceptable y se puede optimizar mediante el uso de cachés, optimización de consultas...</p> <p>Además, las herramientas de este web stack cuentan con una gran cantidad de recursos y documentación disponibles en línea. Así mismo, WAMP está conformado por herramientas de código abierto y gratuitas.</p> <p>Finalmente, las herramientas añadidas como SASS y Gulp permitirán agilizar, optimizar y automatizar el flujo de trabajo ante la realización de tareas repetitivas.</p> <p>¿Cómo funciona este web stack?</p> <p>A continuación se va a describir el flujo de trabajo cuando se realiza una petición web con el WAMP stack:</p> <ol style="list-style-type: none">1. El cliente envía una solicitud de tipo HTTP al servidor de apache.2. Apache procesa la solicitud y determina si se trata de un archivo estático o solicitud dinámica.3. En caso de ser dinámica, se pasa la solicitud al procesador PHP.4. Se ejecuta el código PHP en el servidor y, de ser necesario, se interactúa con la base de datos en MySQL.
--	--

	<ol style="list-style-type: none">5. MySQL retorna los datos solicitados.6. PHP genera el contenido dinámico, combinando la lógica de la aplicación con los datos obtenidos.7. El contenido generado es enviado a Apache.8. Apache envía la respuesta HTTP con el contenido generado al cliente (navegador)9. El navegador renderiza el contenido recibido y es mostrado al usuario.
--	--

Entregar este documento en formato PDF en el TEC Digital según fecha asignada.